A sintaxe mais geral para procedimentos será:

```
procedimento <nome do procedimento> (<lista de parâmetros>);
   <especificação dos parâmetros>
   início
   | <declarações de variáveis locais>
   | C1;
   | C2;
   | .
   | .
   | .
   | Cm;
   fim; {nome do procedimento}
```

A <especificação dos parâmetros> consiste na declaração dos tipos das variáveis que compõem a lista de parâmetros. No exemplo do procedimento TROCA, *X* e *Y* deixariam de ser variáveis globais e passariam a fazer parte da lista de parâmetros:

```
procedimento TROCA (X, Y);
   inteiro: X, Y;
   início
   | inteiro: AUX;
   | AUX ← X;
   | X ← Y;
   | Y ← AUX;
   fim; {TROCA}
```

O algoritmo ficaria:

```
início
|  inteiro: A, B, C, D;
|  procedimento TROCA (X, Y);
|     inteiro: X, Y;
|     início
|     | inteiro: AUX;
|     | AUX ← X;
|     | X ← Y;
|     | Y ← AUX;
|     fim; {TROCA}
|  leia (A, B);
|  imprima (A, B);
|  TROCA (A, B);
|  imprima (A, B);
|  leia (C, D);
|  imprima (C, D);
|  TROCA (C, D);
|  imprima (C, D);
fim.
```


A utilização de procedimentos permite que um conjunto de comandos usado repetidas vezes em vários pontos do algoritmo (ou de uso geral) possa ser escrito uma única vez (na declaração) e chamado o número de vezes necessárias, desde que suficientemente parametrizados.

Exemplo:

```
procedimento ABS (X, Y); {cabeçalho}
   real: X, Y; {especificação dos parâmetros formais}
   início {corpo do procedimento}
   |      {não tem variável local}
   |      se X ≥ 0 então Y ← X;
   |      |        senão Y ← – X;
   |      fim se;
   fim; {ABS}
```

No exemplo anterior, *X* é um parâmetro de *entrada* e *Y*, um parâmetro de *saída*. *X* e *Y* são denominados *parâmetros formais*.

Como podemos ter no algoritmo diversas chamadas do procedimento ABS para encontrar o valor absoluto de diferentes variáveis, os parâmetros que aparecem na chamada do procedimento são denominados *parâmetros efetivos*. No momento da chamada, quando então o procedimento vai ser executado, os parâmetros formais são substituídos pelos parâmetros efetivos respectivos. Um parâmetro de saída deve ser, necessariamente, um identificador, enquanto que um parâmetro de entrada pode ser um identificador, uma constante ou uma expressão aritmética.

Exemplos ilustrativos

Exemplo 1:

```
início
|  real: A, B, C, D, E;
|  procedimento ABS (X, Y);
|     real: X, Y;
|     início
|     | se X ≥ Ø
|     | |  então Y ← X;
|     | |  senão Y ← – X;
|     | fim se;
|     fim; {ABS}
|  A ← 5; B ← – 3; E ← – 10;
|  ABS (A, C);
|  imprima (A, C);
|  ABS (– 3, D)2
|  imprima ("– 3", D)
|  ABS (E ** 3, D);
|  E ← D * A + B;
|  imprima (E);
|  ABS (A, D);
|  ABS (D, C);
|  E ← C * D;
|  imprima (E);
fim.
```